CLAIMS

What is claimed is:

1. A data scrambler for a high density optical recording/reproducing apparatus, the data scrambler comprising:

a random data generator which generates random data having a random data generation cycle based on a result obtained by multiplying at least a size of a first data frame by a result obtained by dividing a data amount of two tracks in an outermost circumference of the optical disc by a size of a second data frame.

2. The data scrambler of claim 1, wherein the size of the first data frame is one sector, and the size of the second data frame is one error correction block.

3. The data scrambler of claim 1, wherein the random data generation cycle is at least as great as the result obtained by multiplying at least the size of the first data frame by the result obtained by dividing the data amount of the two tracks in the outermost circumference of the optical disc by the size of the second data frame.

4. The data scrambler of claim 1, wherein the random data generator comprises:

registers, serially arranged, which shift-store *n* bits and generate random data, and uses a total of *n* values as initial values, including a first initial value, first register values, which are output after shifting the first initial value 7 times, a second initial value immediately after a capacity required for return of the first initial value and the first register values, and second register values which are output after shifting the second initial value 7 times; and

a first serial logic circuit having a plurality of logic gates, which exclusive-OR outputs ones of the registers which correspond to a number of effective branches with a predetermined branch value, and outputs of neighboring ones of the logic gates are fed back to a least significant one of the registers,

wherein the data scrambler further comprises a second logic circuit which scrambles outputs of a predetermined number of least significant ones of the registers and input data in units of byte.

5. The data scrambler of claim 4, wherein the random data generation cycle is 2^{16} (=64 K) when n is 16.

6. The data scrambler of claim 4, wherein the number of effective branches is at least 4, and the effective branch value is any one of 8016h, 801Ch, 8029h, 80D0h, 810Ah, 810Ch, 8112h, 8142h, 8148h, 8150h, 8214h, 8241h, 8244h, 8248h, 8260h, 8320h, 8406h, 8430h, 8540h, 8580h, 8610h, 8805h, 8821h, 8841h, 8842h, 8920h, 8940h, 8A04h, 9028h, 9082h, 9120h, 9420h, 9840h, 9C00h, A084h, A101h, A108h, A140h, A440h, A801h, A840h, B010h, B400h, C009h, C00Ah, C042h, C108h, C120h, C208h, C801h, CA00h, and D008h.

7. The scrambler of claim 4, wherein the effective branch value is "B400h" and the initial values include the first initial value as 0001h, the first register values as (0002h, 0004h, 0008h, 0010h, 0020h, 0040h, 0080h), which are obtained by left-shifting 0001h 7 times, the second initial value as 7E80h, a result of the registers after 32K, which is the capacity required for the return of the first initial and the first register values (B400h, 0002h, 0004h, 0008h, 0010h, 0020h, 0040h, 0080h), and the second register values as (FF01h, FE02h, FC04h, F808h, F011h, E023h, C046h), which are obtained by left-shifting the second initial value 7E80h 7 times.

8. The data scrambler of claim 1, wherein the random data generator comprises:

registers, serially arranged, which shift-store *n* bits and generate random data, and use a total of n values as initial values, including a first initial value and register values, which are supplied in each 4K times left-shifting of the first initial value,

a first serial logic circuit having a plurality of logic gates, which exclusive-ORs outputs of ones of the registers which correspond to a number of effective branches with a

predetermined branch value, and outputs of neighboring ones of the logic gates are fed back to a least significant one of the registers,

wherein the data scrambler further comprises a second logic circuit which scramblers outputs of a predetermined number of least significant ones of the registers and input data in units of byte.

9. The data scrambler of claim 8, wherein the random data generation cycle is 2^{16} (=64 K) when n is 16.

10. The data scrambler of claim 8, wherein the number of effective branches is at least 4, and the effective branch value is any one of 8016h, 801Ch, 8029h, 80D0h, 810Ah, 810Ch, 8112h, 8142h, 8148h, 8150h, 8214h, 8241h, 8244h, 8248h, 8260h, 8320h, 8406h, 8430h, 8540h, 8580h, 8610h, 8805h, 8821h, 8841h, 8842h, 8920h, 8940h, 8A04h, 9028h, 9082h, 9120h, 9420h, 9840h, 9C00h, A084h, A101h, A108h, A140h, A440h, A801h, A840h, B010h, B400h, C009h, C00Ah, C042h, C108h, C120h, C208h, C801h, CA00h, and D008h.

11. The data scrambler of claim 8, wherein the value of the effective branch is "B400h" and the initial values include the first initial value as 0001h and the register values as (3DADh, D4E7h, FDCAh, EBCCh, 292Eh, 50F0h, BFCAh, 7F80h, D36Eh, BB39h, 5DFFh, A809h, 6647h, 8044h, 0304h), which are obtained after every 4096 times left-shifting of the first initial value 0001h.

12. The data scrambler of claim 1, wherein the random data generator changes the effective branch value in units of a first cycle and generates a second cycle of the random data according to a control value.

13. The data scrambler of claim 12, wherein the first cycle corresponds to an error correction block and the second cycle corresponds to a sector.

14. The data scrambler of claim 12, wherein the second cycle is 4 K, and the control value is one of 829h, 834h, 84Ch, 868h, 883h, 891h, 8B0h, 8C2h, 906h, 960h, 990h, A03h, A18h, B04h, C48h, and CA0h in units of an error correction block.

15. The data scrambler of claim 12, wherein the random data generator comprises:

a decoder which supplies 12 output bits, which correspond to 16 kinds of control values, in units of an error correction block;

registers, arranged serially, which shifting-store 12 bits and generate random data in units of a sector;

a selection output circuit which receives the 12 output bits supplied from the decoder, as a selection signal, supplies a predetermined value for each bit of an effective branch among the 12 output bits from the decoder, and otherwise supplies corresponding outputs of the registers, to generate 12 outputs; and

a first logic circuit which exclusive-ORs the 12 output bit of the selection output circuit and the 12 output bits of the registers and then, feeds back a result of the exclusive-ORing only for each bit of the effective branch among the 12 output bits from the decoder,

wherein the data scrambler further comprises a second logic circuit which scrambles outputs of a predetermined number of least significant ones of the registers and input data in units of a byte.

16. The data scrambler of claim 15, wherein initial values of the register are newly set in each error correction block.

17. A data scrambling method using a random data generator for a high density optical recording/reproducing apparatus using an optical disc, the data scrambling method comprising:

generating random data having a random data generation cycle based on a result by multiplying at least a size of a first data frame by a result, which is obtained by dividing a data

amount of two tracks in an outermost circumference of the optical disc by a size of a second data frame.

18. The data scrambling method of claim 17, wherein the size of the first data frame is a sector, and the size of the second data frame is an error correction block.

19. The data scrambling method of claim 17, wherein the random data generation cycle is at least as great as the result obtained by multiplying at least the size of the first data frame by the result obtained by dividing the data amount of the two tracks in the outermost circumference of the optical disc by the size of the second data frame.

SUB A1

20. The data scrambling method of claim 17, wherein the generating of the random data comprises shift-storing, *n* bits in registers and then, generating random data, wherein a total of *n* values are used as initial values, including a first initial value, first register values, which are obtained by shifting the first initial value 7 times, a second initial value immediately after a capacity required for return of the first initial value and the first register values, and second register values, which are obtained by shifting the second initial value 7 times,

wherein the data scrambling method further comprises exclusive-ORing outputs of a predetermined number of least significant ones of the registers and input data in units of byte .

21. The data scrambling method of claim 17, wherein the generating of the random data comprises shift-storing *n* bits in registers and then, generating random data, wherein a total of n values are used as initial values, including a first initial value and register values which are supplied in each 4K times left-shifting of the first initial value;

wherein the data scrambling method further comprises exclusive-ORing outputs of a predetermined number of least significant ones of the registers and input data in units of bit.

22. The data scrambling method of claim 17, wherein the generating of the random data comprises newly setting initial values in units of an error correction block of registers generating random data in units of a sector, corresponding to 16 kinds of control values supplied in units of the error correction block,

wherein the data scrambling method further comprises exclusive-ORing outputs of a predetermined number of least significant ones of the registers and input data in units of byte .

23. The data scrambler of claim 1, wherein:

the first data frame is approximately 4Kb (Kilobytes) in size;

the second data frame is approximately 64Kb in size; and

the data amount of the two tracks in the outermost circumference of the optical disc is approximately 568Kb.

24. The data scrambler of claim 1, wherein:

the first data frame is approximately 8Kb (Kilobytes) in size;

the second data frame is approximately 64Kb in size; and

the data amount of the two tracks in the outermost circumference of the optical disc is approximately 568Kb.

25. The data scrambler of claim 1, wherein:

the first data frame is approximately 4Kb (Kilobytes) in size;

the second data frame is approximately 128Kb in size; and

the data amount of the two tracks in the outermost circumference of the optical disc is approximately 568Kb.

26. The data scrambler of claim 1, wherein the random data generation cycle is at least 64Kb (Kilobytes).

27. The data scrambler of claim 4, wherein:

the size of the first data frame is a sector and the size of the second data frame is an error correction block; and

the initial values are determined by an upper 4 bits of a last byte in a 4-byte identification code which is allocated in each of a plurality of the first data frames.

28. The data scrambler of claim 8, wherein:

the size of the first data frame is a sector and the size of the second data frame is an error correction block; and

the initial values are determined by an upper 4 bits of a last byte in a 4-byte identification code which is allocated in each of a plurality of the first data frames.

29. The data scrambler of claim 15, wherein:

the size of the first data frame is a sector and the size of the second data frame is an error correction block; and

the initial values are determined by an upper 4 bits of a last byte in a 4-byte identification code which is allocated in each of a plurality of the first data frames.

30. The data scrambler of claim 1, wherein the random data generator:

a decoder to selectively output n bits as valid and invalid bits in response to input m bits;

n registers arranged in serial, which shift and store the n bits, to generate shifted n bits as the random data;

a selection circuit which selects a predetermined value or the shifted n bits for ones of the shifted n bits, to generate a selection signal; and

logic gates arranged in serial, which perform XOR operations on the ones of the shifted n bits, the ones of the shifted n bits, and an output of an adjacent more significant one of the logic circuits, wherein the output of the logic gate associated with a least significant of the ones of the shifted n bits is fed back to a least significant one of the registers.

31. The data scrambler of claim 30, further comprising:

a scrambling circuit which performs XOR operations on a plurality of least significant ones of the shifted n bits and corresponding input data bits after the n registers 8-bit left shift the n bits.

32. The data scrambler of claim 30, further comprising:

a scrambling circuit which performs XOR operations on a plurality of least significant ones of the shifted n bits and corresponding input data bits after each one-bit left shift of the n-registers.

33. The data scrambler of claim 30, further comprising:

a scrambling circuit which performs XOR operations on a plurality of least significant ones of the shifted n bits and corresponding input data bits after the n registers 4k left shift the n bits.

34. The data scrambling method of claim 20, wherein:

the size of the first data frame is a sector and the size of the second data frame is an error correction block; and

the method further comprising determining the initial values by an upper 4 bits of a last byte in a 4-byte identification code which is allocated in each of a plurality of the first data frames.

35. The data scrambling method of claim 21, wherein:

the size of the first data frame is a sector and the size of the second data frame is an error correction block; and

the method further comprising determining the initial values by an upper 4 bits of a last byte in a 4-byte identification code which is allocated in each of a plurality of the first data frames.

36. The data scrambling method of claim 17, wherein the generating of the random data comprises shift-storing n bits in registers and then, generating random data, wherein a total of n values are used as initial values, including a first initial value and register values, which are supplied in each 4K left-shifting of the first initial value;

wherein the data scrambling method further comprises supplying exclusive-ORing outputs of a predetermined number of least significant ones of the registers and input data in units of byte.

37. The data scrambling method of claim 17, wherein the generating of the random data comprises:

selectively outputting n bits as valid and invalid bits in response to input m bits;

shifting and storing the n bits in serially arranged registers, to generate shifted n bits as the random data;

selecting a predetermined value or the shifted n bits for ones of the shifted n bits, to generate a selection signal; and

performing XOR operations on the ones of the shifted n bits, the ones of the shifted n bits, and an output of an adjacent more significant one of the logic circuits, and feeding back the output associated with a least significant of the ones of the shifted n bits to a least significant one of the registers.

38. The data scrambling method of claim 37, further comprising:

performing XOR operations on a plurality of least significant ones of the shifted n bits and corresponding input data bits after 8-bit left-shifting the n bits in the n registers.

39. The data scrambling method of claim 37, further comprising:

performing XOR operations on a plurality of least significant ones of the shifted n bits and corresponding input data bits after one-bit left shifting of the n bits in the n registers; and

repeating the performing of the XOR operations on the plurality of least significant ones of the shifted n bits and the corresponding input data bits after each of repeated one-bit left shifting of the n bits in the n registers.

40. The data scrambling method of claim 37, further comprising:

performing XOR operations on a plurality of least significant ones of the shifted n bits and corresponding input data bits after left shifting 4K times the n bits in the n registers.

41. A data scrambler for a high density optical recording and/or reproducing apparatus using an optical disc, comprising:

a random data generator which generates random data and adjusts a random data generation cycle of the random data based upon a data amount of two tracks in an outermost circumference of the optical disc; and

a scrambling circuit to scramble the random data.

42. The data scrambler of claim 41, wherein the optical disc has error correction blocks each comprising sectors, wherein:

the random data generator adjusts the random data generation cycle of the random data based upon the size of each sector and a size of each error correction block.

43. A data scrambler for a high density optical recording and/or reproducing apparatus using an optical disc, comprising:

a random data generator which generates random data and adjusts a random data generation cycle of the random data based upon a data amount in an innermost circumference of the optical disc; and

a scrambling circuit to scramble the random data.

44. The data scrambler of claim 43, wherein the optical disc has error correction blocks each comprising sectors, wherein:

the random data generator adjusts the random data generation cycle of the random data based upon the size of each sector and a size of each error correction block.

45. A data scrambler for a high density optical recording and/or reproducing apparatus using an optical disc having error correction blocks each comprising sectors, the data scrambler comprising:

a random data generator which generates random data and adjusts a random data generation cycle of the random data based upon a size of each sector and a size of each error correction block; and

a scrambling circuit to scramble the random data.

46. A data scrambler for a high density optical recording and/or reproducing apparatus using an optical disc having second data frames each comprising a plurality of first data frames, the data scrambler comprising:

a random data generator which generates random data and adjusts a random data generation cycle of the random data based upon a size of each first data frame and a size of each second data frame; and

a scrambling circuit to scramble the random data.